SCI & OSL Technical Excellence Symposium 09
The Syslinux Project

Overview of a modern bootloader

Erwan Velu (erwanaliasr1@gmail.com)
Bootloaders aren't dead!

- Bootloaders are running on our systems for ages
- Lilo & Grub are, by far, the most known/popular
- But it looks like they didn't evolved since the dinosaur era
- "Their development is complete, not need to spend time on it"
- 'The Syslinux Project' has never been such active since the beginning (15 years ago)
- Many contributors joined the project since the past 2 years
- Why such activity? Bootloaders aren't such sexy....
What’s The Syslinux Project?

- A collection of bootloaders
  - ISOLINUX for booting CDROMs (El Torito)
  - SYSLINUX for booting from a FAT filesystem
  - EXTLINUX for booting from an EXT3 filesystem
  - PXELINUX for booting over the network (PXE)

- Aims at booting systems from every possible media

- Developed since 1994 by Hans Peter Anvin (HPA)
  - Historical Linux Kernel Hacker
  - Co-maintainer of the x86 code
  - Co-webmaster of Kernel.org
What’s The Syslinux Project?

'The Syslinux Project' is made of

- A core layer
  - Low level operations
  - Memory management
  - Written in Assembly

- A com32 module framework
  - Easy-to-develop extensions
  - Written in C
  - Many libraries as helper
    - libjpg, zlib, pci, dmi, ...
COM32, a modular approach

COM32 modules have crystallized contributions/contributors as

- No ASM skills required to add new features or bugfix
- High level API / libraries
- Everyone who knows C can contribute!
- Nicest COM32 modules
  - Menu / cmenu / vesamenu
  - Chainloading / ifcpu
  - Gfxboot (in progress) / rosh
  - Mboot / HDT
  - Lua scripting (to come)
Zooming on PXElinux

- PXE (Pre-Execution Environment) make systems booting on the LAN
  - Requires some
    - infrastructure (DHCP / TFTP)
    - A PXE-compliant Ethernet device
    - A bootstrap to execute

- PXElinux is *the* PXE bootstrap
  - Downloaded by the PXE ROM and then executed locally
  - It uses ROM’s interface to use the network (no driver in PXElinux)
  - Have to load & execute *something* executable, typically a kernel
    - Memdisk is a special kernel to load disk images (floppy/HDD/iso)

- PXE and so PXElinux can only load data from the TFTP service
  - Very, very limiting… (slow, insecure, LAN bounded, ….)
gPXE, the *Waow* effect

- Etherboot project is known for building PXE ROM images
- Their newest project change the rules
- gPXE is a 100% OSS implementation of a PXE ROM
- In addition of the PXE specification, it adds many new ‘interfaces’
  - TFTP is no more alone!
  - FTP, HTTP, HTTPS, NFS, ISCSI, … are new friends
- Beginning of a new booting era
  - No more LAN restriction
  - Booting content can be generated dynamically (cgi, …)
  - Let’s boot Windows 2008 server from an iscsi target
    - That’s not new isn’t? Except it just uses regular hardware!
  - Int13h emulation, why not booting Windows 3.1 via ISCSI ?!?
gPXElinux, a less intrusive transition

- gPXE is awesome but it has a huge drawback
- gPXE have to be flashed in place of the regular PXE ROM
  - Could be difficult on embedded systems as part of the BIOS
  - What about the warranty in case of failure?
- While waiting for a broader adoption, we have a solution
- gPXElinux = gPXE + PXElinux
  - The original PXE ROM is loading gPXElinux.0 via TFTP
  - gPXElinux.0 installs itself in memory in place of the actual PXE ROM
  - gPXElinux.0 starts the built-in pxelinux.0
  - PXElinux is now loaded and has the benefits of the additional features of gPXE
  - gPXElinux can load FTP, HTTP, ISCSI content on a un-modified hardware setup
Boot Kernel.org, the universal boot

- And if … « We could have a centralized repository for booting? »

- Systems usually have PXE + some LAN / Internet connection

- Why should I have to
  - Go to a project site
  - Download some bootable content
  - Burn an image
  - Boot on it
  - Throw away the bootable media

- BKO service offer a menu of bootable content
  - Linux installers
  - Linux LiveCDs
  - Debugging Tools (HDT, Pxeknife, …)
What’s next?

- Syslinux 4.x is currently under alpha
  - File system API in C
  - COM32 Relocatable
  - GFXBOOT as com32

- Syslinux 5.x
  - ELF Linker
  - Native TCP for PXELINUX (lwip)
  - End of support for COM16 modules

- Syslinux 6.x
  - EFI support
Conclusion

- Syslinux offer via its suite, innovative solutions to boot your system
- Can use any kind of bootable device to start your system
  - CDROM / USB / Network / Local drive
- Can be easily extended via its COM32 framework
- Can use the benefits of gPXE to extend booting options
  - Booting over Internet!

Bootloaders aren’t dead! Let’s innovate!

IRC: #hdt, #syslinux on freenode
Mailing lists: hdt@zytor.com, syslinux@zytor.com